

~~logical primary key being created in a physical primary key of the versioned table];~~

defining to said database management system at least a second portion [~~one column~~] of the physical primary key as a version effective reference value;

deriving version differentiation criteria information from a version differentiation predicate included in a request submitted by a database user, the version differentiation predicate identifying [~~including a name of~~] the versioned table and [~~defined to a database,~~] a target effective status[, ~~and a target value for version differentiation processing~~]; and

retrieving rows of the versioned table that satisfy the version differentiation criteria as a function of [~~derived from~~] the version differentiation predicate submitted by the database user and the version effective reference value defined to said database management system, including [by] comparing the version effective reference value[s] of the versioned table as defined to the database management system with the version differentiation criteria submitted by the database user.

2. (Amended) The method for row version differentiation of Claim 1 wherein said version effective reference value is a version effective start value, the method for version differentiation further comprising:

identifying [~~of~~] a version effective end value that does not participate in said physical primary key of said versioned table;

said retrieving of rows from the versioned table including comparing the effective end values of the versioned table with the version differentiation criteria.

3. (Amended) The method for row version differentiation of Claim 2 further comprising:

defining an effective window for each row of the versioned table as a function of the effective start value and the version effective end value for each row in the versioned table; and

validating the effective window for one row of the versioned table to ensure that the effective window for the one row of the versioned table does not overlap with effective windows for other rows of the versioned table having logical primary keys matching the logical primary key for the one row of the versioned table.

4. (Amended) The method for row version differentiation of Claim 1 further comprising:

identifying to said database management system a referential constraint specifying as a parent said versioned table; and

ensuring that rows exist in the versioned table such that the values of their logical primary keys correspond to the values of the columns of a dependent table identified in the referential constraint for an existing row of the dependent table.

5. (Amended) The method for row version differentiation of Claim 4 wherein said version effective reference value is a version effective start value, the method for version differentiation further including:

identifying a column [row] of the dependent table during the definition of said referential constraint for use as a referential constraint effective start value; and

comparing said referential constraint effective start value and said versioned effective start value.

6. (Amended) The method for row version differentiation of Claim 5 further including:

identifying a column [row] of the dependent table, during the definition of said referential constraint, for use as a referential constraint effective end value; and

comparing said referential constraint effective start value to said versioned effective start value and said effective end value.

7. (Amended) The method for row version differentiation of Claim 6 further comprising:

defining a referential constraint effective window for each row of the dependent [versioned] table as a function of the referential constraint effective start value and the referential constraint effective end value for each row of the dependent [versioned] table; and

validating the referential constraint effective window for one row of the dependent [versioned] table to ensure that the referential constraint effective window for the one row of the dependent [versioned] table does not overlap with the referential constraint effective windows for other rows of the dependent [versioned] table having logical primary keys matching the logical primary key for the one row of the dependent [versioned] table.

8. (Amended) A method for row version differentiation in a database management system comprising:

identifying a versioned table to said database management system;

creating a physical primary key of the versioned table;

defining to said database management system at least a first portion of said physical primary key as a logical primary key [~~creating a logical primary key, comprising a prescribed number of columns in the versioned table, the logical primary key being created in a physical primary key of the versioned table~~];

defining to said database management system at least a second portion [~~one column~~] of the physical primary key as a version effective reference value;

deriving version differentiation criteria information from a version differentiation predicate included in a request submitted by a database user, the version differentiation predicate identifying [~~including a name of~~] the versioned table and [~~defined to a database,~~] a target effective status, and a target value range as defined by a target start value and a target end value that are included in said version differentiation predicate; and

retrieving rows of the versioned table that satisfy the version differentiation criteria as a function of [~~derived from~~] the version differentiation predicate submitted by the user and the version effective reference value defined to said database management system, including [by] comparing the version effective reference value[s] of the versioned table as defined to the database management system with the version differentiation criteria submitted by the database user.

9. (Amended) The method for row version differentiation of Claim 8 further comprising:

validating said target value range for one row of the versioned table to ensure that the target value range for the one row of the versioned table does not overlap with the target value ranges for other rows of the versioned table having logical primary keys matching the logical primary key for the one row of the versioned table.

REMARKS

The specification and claims have been amended to correct minor informalities and to address other issues raised by the Examiner in the office action and during the December 10, 1998 telephonic conference. Claims 1 through 9 have been amended. Nine claims remain pending in the application: Claims 1 through 9. Reconsideration of Claims 1 through 9 in

view of the amendments above and arguments below is respectfully requested.

At the outset, Applicant thanks the Examiner for his willingness and availability to meet telephonically on December 10, 1998. Applicant felt the Examiner's comments were very helpful and appreciates the Examiner's indication that the proposed claim amendments overcome the rejections addressed below.

Turning to the specific objections and rejections:

1. Claims 3 and 7 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form, including all of the limitations of the base claim and any intervening claims. As discussed below, it is respectfully submitted that the rejections are overcome. Thus, Claims 3 and 7 are dependent on allowable claims and the objection is thus overcome.

2. Claims 1, 2, 4, 5, 6, 8 and 9 stand rejected under 35 U.S.C. § 103(a), as being unpatentable over U.S. Patent No. 5,594,899 (Knudsen) in view of U.S. Patent No. 5,594,899 (Lorie).

With respect to independent Claim 1, Knudsen teaches an object oriented database. In all material respects (i.e., relating to the present application) Knudsen teaches an entirely conventional database management system.

Lorie teaches a method of concurrency control that overcomes lock delays with concurrent transactions and query processing. Tables of a database in accordance with the teachings of Lorie are copied so that multiple versions of the tables can each be updated concurrently without having to wait for a concurrent query to finish. The copies of the tables are copied back into the database as a function of a Concurrency Control Number similar to a transaction serial number. The concurrency control number for each copy of the

A

table is unique, and is discarded after the table is closed by the user and copied back into the database.

A combination of Knudsen and Lorie would provide a system that would add the concurrency control of Lorie to the Knudsen object oriented database.

In contrast, Applicant creates versions of rows that contain version effective reference values as a portion of their primary key. The version effective reference value indicates when the row is "effective". Applicant's versioned rows are enduring entities created for the purpose of defining past, present, and/or future effective windows for the information contained in the versioned rows. Knudsen does not teach versioned tables (as pointed out by the Examiner). The concurrency control numbers in Lorie are ephemeral entities created by the database management system, and unseen and inaccessible by the user. The version information employed by Lorie is discarded when the copy is written to the database, and is not, therefore, created as a portion of a primary key.

Thus, Applicant's Primary Key (PK) has two subsets; a Logical Primary Key (LPK), which is similar to the Primary Key in Knudsen and Lorie, and a Version Effective Reference Value (VERV), which neither Knudsen nor Lorie has, which serves to differentiate between versioned rows of similar data. Neither Knudsen nor Lorie break their Primary Key into subsets, let alone break their Primary Key into subsets for the purpose of version differentiation.

Advantageously, in accordance with Applicant's invention, the user is able to retrieve versioned information by inputting version differentiation criteria containing a predicate that indicates the versioned table, a target value for version differentiation processing, and a target effective status to determine which version of the row is retrieved. Neither Knudsen nor Lorie permits this functionality. Instead, Knudsen and Lorie require that version differentiation criteria be embodied in complex code

Amendment
S.N. 08/790,165
Page 8

algorithms, which is precisely the problem sought to be, and in fact overcome, by Applicant's invention as claimed.

Thus, the embodiment claimed in Claim 1, as discussed with the Examiner during our recent telephone conference, is vastly different from that which is shown or suggested by the cited patents either singly or in combination. Thus, it is respectfully submitted that the Examiner's rejection is overcome and that Claim 1 is allowable.

With respect to independent Claim 8, every element is the same as Claim 1 except that the version differentiation predicate entered by the user includes a target value range containing a target start value and a target end value, instead of a single target value. Thus, it is respectfully submitted that the rejection is overcome and that independent Claim 8 is allowable.

With respect to dependent Claims 2, 4, 5, 6, and 9, Applicant has shown above that the cited references do not render the subject matter of Claims 1 or 8 obvious. Therefore, Applicant has also shown that dependent Claims 2, 4, 5, 6, and 9 should be allowable and thus respectfully requests their allowance for the same reasons as Claims 1 and 8.

Applicant acknowledges with appreciation the Examiner's indication that Claims 3 and 7 would be allowable if rewritten in independent form.

Applicant acknowledges with appreciation the Examiner's indication that Claims 1 through 9 may be allowable if amended as discussed during the telephonic conference of December 10, 1998.

By way of this amendment, Applicant has made a diligent effort to place the claims in condition for allowance. However, should there remain any outstanding issues

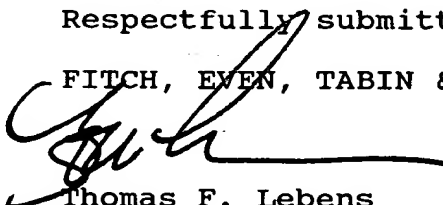
Amendment
S.N. 08/790,165
Page 9

that require adverse action, it is respectfully requested that Examiner telephone the undersigned at (619) 552-1311 so that such issues may be resolved as expeditiously as possible.

In view of the above, Applicant submits that Claims 1 through 9 are now in condition for allowance, and prompt and favorable action is earnestly solicited.

Respectfully submitted,

FITCH, EVEN, TABIN & FLANNERY



Thomas F. Lebens
Reg. No. 38,221

Dated: 6-10-99, 1999

Address all correspondence to:
Thomas F. Lebens
FITCH, EVEN, TABIN & FLANNERY
120 So. LaSalle Street, Suite 1600
Chicago, Illinois 60603
619) 552-1311